```php
<?php

/**
 * @author GOLAY Brian
 * @version 1.0 (2021/05/20)
 * Flight-related functions
 */

namespace FlightClub\sql;

use FlightClub\sql\DBConnection;

require_once 'dbConnection.php';

/**
 * Flight related class that contain flight related functions with database interraction
 */
class FlightDAO
{

    /**
     * Get all flight of an user with his id
     *
     * @param int $idUser
     * @return array[mixed]
     */
    public static function getAllUserFlightByUserId($idUser)
    {
        $db = DBConnection::getConnection();
        $sql = "SELECT * FROM `tbl_flight` join `tbl_user-flight` on `tbl_flight`.`Id_Flight` = `tbl_user-flight`.`Id_Flight` WHERE `Id_User` = :idUser";
        $request = $db->prepare($sql);
        $request->execute([
            ':idUser' => $idUser
        ]);
        return $request->fetchAll();
    }

    /**
     * Get all flight of an user with his id with only the data needed for the PDF
     *
     * @param int $idUser
     * @return array[mixed]
     */
    public static function getAllUserFlightByUserIdForExport($idUser)
    {
        $db = DBConnection::getConnection();
        $sql = "SELECT `Cd_Role`, `No_Flight`, `Dttm_Departure`, `Dttm_Arrival`, `Tm_Engine_On`, `Tm_Engine_Off`, `Nm_Plane`, `No_Plane`, `Cd_ICAO_Departure`, `Cd_ICAO_Arrival`, `Cd_Flight_Type`, `Cd_Flight_Mode`, `Nb_Passengers` FROM `tbl_flight` join `tbl_user-flight` on `tbl_flight`.`Id_Flight` = `tbl_user-flight`.`Id_Flight` WHERE `Id_User` = :idUser";
        $request = $db->prepare($sql);
        $request->execute([
            ':idUser' => $idUser
        ]);
        return $request->fetchAll();
    }

    /**
     * Add a flight to the database
     *
     * @param int $idFlight
     * @param int $idUser
     * @param string $role
     * @param string $flight
     * @param string $dateDeparture
     * @param string $dateArrival
     * @param string $timeDeparture
     * @param string $timeArrival
     * @param string $timeEngineOn
     * @param string $timeEngineOff
     * @param string $typeAircraft
     * @param string $registrationPlane
     * @param string $icaoDeparture
     * @param string $icaoArrival
     * @param string $flightType
     * @param string $flightMode
     * @param string $weather
     * @param int $passengers
     * @param string $notes
     * @return void
     */
    public static function addFlight(
        $idFlight,
        $idUser,
        $role,
        $flight,
        $dateDeparture,
        $dateArrival,
        $timeEngineOn,
        $timeEngineOff,
        $typeAircraft,
        $registrationPlane,
        $icaoDeparture,
        $icaoArrival,
        $flightType,
        $flightMode,
        $weather,
        $passengers,
        $notes
    ) {

        $db = DBConnection::getConnection();
        $sql = "INSERT INTO `tbl_flight`(`Id_Flight`, `No_Flight`,`Dttm_Departure`,`Dttm_Arrival`,
        `Tm_Engine_On`,`Tm_Engine_Off`,`Nm_Plane`,`No_Plane`,`Cd_ICAO_Departure`,
         `Cd_ICAO_Arrival`,`Cd_Flight_Type`,`Cd_Flight_Mode`,`Txt_Meteo`,`Nb_Passengers`,`Txt_Note`)
    VALUES (:idFlight, :flight, :dateDeparture, :dateArrival, :timeEngineOn,
            :timeEngineOff, :typeAircraft, :registrationPlane, :icaoDeparture, :icaoArrival, :flightType, :flightMode, :weather, :passengers, :notes);
            INSERT INTO `tbl_user-flight`(`Id_User`, `Id_Flight`, `Cd_Role`)
    VALUES (:idUser, :idFlight, :role)";
        $request = $db->prepare($sql);
        $request->execute([
            ':idFlight' => $idFlight,
            ':idUser' => $idUser,
            ':role' => $role,
            ':flight' => $flight,
            ':dateDeparture' => $dateDeparture,
            ':dateArrival' => $dateArrival,
            ':timeEngineOn' => $timeEngineOn,
            ':timeEngineOff' => $timeEngineOff,
            ':typeAircraft' => $typeAircraft,
            ':registrationPlane' => $registrationPlane,
            ':icaoDeparture' => $icaoDeparture,
            ':icaoArrival' => $icaoArrival,
            ':flightType' => $flightType,
            ':flightMode' => $flightMode,
            ':weather' => $weather,
            ':passengers' => $passengers,
            ':notes' => $notes
        ]);
    }


/**
     * Edit a flight to the database
     *
     * @param int $idFlight
     * @param int $idUser
     * @param string $role
     * @param string $flight
     * @param string $dateDeparture
     * @param string $dateArrival
     * @param string $timeEngineOn
     * @param string $timeEngineOff
     * @param string $typeAircraft
     * @param string $registrationPlane
     * @param string $icaoDeparture
     * @param string $icaoArrival
     * @param string $flightType
     * @param string $flightMode
     * @param string $weather
     * @param int $passengers
     * @param string $notes
     * @return void
     */
    public static function editFlight(
        $idFlight,
        $idUser,
        $role,
        $flight,
        $dateDeparture,
        $dateArrival,
        $timeEngineOn,
        $timeEngineOff,
        $typeAircraft,
        $registrationPlane,
        $icaoDeparture,
        $icaoArrival,
        $flightType,
        $flightMode,
        $weather,
        $passengers,
        $notes
    ) {

        $db = DBConnection::getConnection();
        $sql = "UPDATE `tbl_flight` SET `No_Flight` = :flight,`Dttm_Departure` = :dateDeparture, `Dttm_Arrival` = :dateArrival,
        `Tm_Engine_On` = :timeEngineOn,`Tm_Engine_Off` = :timeEngineOff,`Nm_Plane` = :typeAircraft,`No_Plane` = :registrationPlane, `Cd_ICAO_Departure` = :icaoDeparture,
        `Cd_ICAO_Arrival` = :icaoArrival,`Cd_Flight_Type` = :flightType,`Cd_Flight_Mode` = :flightMode,`Txt_Meteo` = :weather,`Nb_Passengers` = :passengers,`Txt_Note` = :notes WHERE `tbl_flight`.`Id_Flight` = :idFlight;
        UPDATE `tbl_user-flight`SET `Cd_Role` = :role WHERE `Id_Flight` = :idFlight)";
        $request = $db->prepare($sql);
        $request->execute([
            ':idFlight' => $idFlight,
            ':idUser' => $idUser,
            ':role' => $role,
            ':flight' => $flight,
            ':dateDeparture' => $dateDeparture,
            ':dateArrival' => $dateArrival,
            ':timeEngineOn' => $timeEngineOn,
            ':timeEngineOff' => $timeEngineOff,
            ':typeAircraft' => $typeAircraft,
            ':registrationPlane' => $registrationPlane,
            ':icaoDeparture' => $icaoDeparture,
            ':icaoArrival' => $icaoArrival,
            ':flightType' => $flightType,
            ':flightMode' => $flightMode,
            ':weather' => $weather,
            ':passengers' => $passengers,
            ':notes' => $notes
        ]);
    }



    /**
     * Get all flight order by their id descending
     *
     * @return array[mixed]
     */
    public static function getAllFlightOrderById()
    {
        $db = DBConnection::getConnection();
        $sql = "SELECT * FROM `tbl_flight` ORDER BY `Id_Flight` desc";
        $request = $db->prepare($sql);
        $request->execute();
        return $request->fetchAll();
    }


    /**
     * Get a flight with his Id
     *
     * @param int $idFlight
     * @return array[mixed]
     */
    public static function getFlightById($idFlight)
    {
        $db = DBConnection::getConnection();
        $sql = "SELECT * FROM `tbl_flight` WHERE `Id_Flight` = :id";
```

```php
        $request = $db->prepare($sql);
        $request->execute([
            ':id' => $idFlight
        ]);
        return $request->fetch();
    }

}

```